Learning to Align Generative Appearance Priors for Fine-grained Image Retrieval

Shijie Wang[1], **Yadan Luo**[1], **Zijian Wang**[1], **Xin Yu**[2], **Zi Huang**[1]
[1] The University of Queensland, Australia [2] The University of Adelaide, Australia

Abstract

Fine-grained image retrieval (FGIR) typically relies on supervision from seen categories to learn discriminative embeddings for retrieving unseen categories. However, such supervision often biases retrieval models toward the semantics of seen categories rather than the underlying appearance characteristics that generalize across categories, thereby limiting retrieval performance on unseen categories. To tackle this, we propose GAPan, a **G**enerative **A**ppearance **P**rior **a**lignment **n**etwork that reformulates the learning objective from category prediction toward appearance modeling. Technically, GAPan treats retrieval features with an invertible density model based on normalizing flows. In the *forward* direction, the flow maps all instance features into a latent density space, where each seen category is modeled by a class-conditional Gaussian prior and optimized via exact likelihood estimation. This formulation preserves richer appearance details by leveraging the invertible property of the flows. In the *reverse* direction, samples from the high-density regions of these learned priors are mapped back to the feature space to produce appearance-aware anchors that reflect intra-category variation. These anchors supervise a prior-driven alignment objective that aligns retrieval embeddings with category-specific appearance distributions, thereby improving generalization to unseen categories. Evaluations demonstrate that our GAPan achieves state-of-the-art performance on both widely-used fine- and coarse-grained benchmarks.

1 Introduction

Fine-grained image retrieval (FGIR) aims to retrieve images from the same fine-grained subcategory within a common meta-category. It supports applications with large and evolving visual taxonomies, from fine-grained clothing retrieval [1; 21; 40] to ecological monitoring [44; 37; 45]. The difficulty is that the relevant evidence is often subtle and local, while the space of subcategories is naturally open-ended. A retrieval model must therefore go beyond separating the labels observed during training and preserve appearance factors that remain useful for novel subcategories.

Most existing FGIR methods [41; 28; 38; 42; 52] address this challenge through label-driven discriminative learning. Category labels are commonly used to construct discriminative objectives, such as instance pairs, tuples, proxies, or semantic regularizers [6; 14; 32; 48; 36]. These objectives are effective because they directly shape the similarity structure used for retrieval. However, they use labels mainly to specify which training instances should be close or far apart, rather than to model how appearance varies within a fine-grained category. This becomes limiting when the test subcategories differ from the training subcategories. A model may rely on cues that separate seen classes but do not transfer to unseen ones. For example, if training bird species are largely separated by head color, the embedding may overemphasize crown color, whereas novel species with similar heads may require wing patterns or beak shape for retrieval. Thus, category supervision should not only define discriminative relations, but also capture the *distribution* of appearance variations underlying fine-grained similarity.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

As illustrated in Fig. 1, we therefore reinterpret category labels as cues for aggregating instance observations into distribution-level appearance priors. A label alone only specifies category membership; it does not describe which visual cues are reliable or how they vary across instances. We address this by modeling the retrieval features of each category as a class-conditional appearance distribution, rather than a single prototype. This provides a complementary supervisory signal: contrastive learning establishes a retrieval-oriented feature manifold, while appearance priors regularize this manifold with category-level distributional structure.

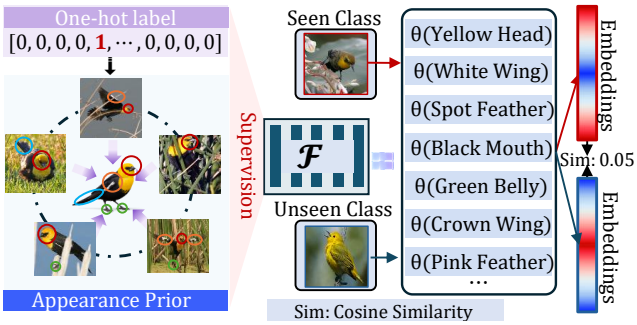

Figure 1: Motivation of GAPan. Beyond one-hot labels, GAPan aggregates diverse instance-level cues into category-specific appearance priors, providing appearance-aware supervision that preserves visual properties and improves generalization to unseen categories.

To instantiate this idea, we introduce GAPan, a Generative Appearance Prior alignment network based on normalizing flows (NF) [19; 23]. GAPan treats retrieval features with an invertible density model: In the *forward* direction, the flow maps retrieval features to a latent density space, where each seen category is modeled by a class-conditional Gaussian prior and optimized by exact likelihood. Because the NF mapping is invertible, this density modeling does not require compressing fine-grained information into a fixed prototype. In the *reverse* direction, samples from high-density regions of the learned priors are mapped back to the retrieval feature space, producing appearance-aware anchors that reflect category-level variation. These anchors are used in a prior-guided alignment objective, which pulls each instance toward anchors from its own category while contrasting it against anchors from other categories. GAPan is trained progressively: the retrieval backbone is first warmed up with an auxiliary contrastive objective to establish a stable retrieval manifold, and is then jointly optimized with likelihood modeling, posterior calibration, and prior-guided alignment. This joint objective converts category labels from discrete indicators of membership into distribution-level supervision over the retrieval space. Compared with single-prototype or purely discriminative supervision, GAPan preserves intra-category appearance variation while encouraging category-level consistency, leading to representations that generalize better to novel fine-grained subcategories.

Our main contributions are summarized below:

- To the best of our knowledge, we are the first to shift supervisory signals from category prediction toward appearance modeling, thereby improving generalization to unseen categories.
- We propose GAPan, an invertible flow framework that distills diverse retrieval features into explicit category-specific appearance priors and leverages samples drawn from their high-density regions as generative supervisory signals for appearance-aware alignment.
- Extensive experiments demonstrate that GAPan achieves state-of-the-art performance on widely used fine-grained and coarse-grained image retrieval benchmarks.

2 Related Work

Fine-grained Image Retrieval. Recent advances in fine-grained image retrieval primarily follow two technical paradigms [53; 24; 16; 28; 52; 12; 51; 39; 35]. Localization-based methods, exemplified by A^2-Net [46], focus on precise object or part localization to enhance retrieval accuracy via reconstruction-based learning. In parallel, metric-based approaches such as DDML [25] learn discriminative embedding spaces through tailored distance metrics, while NIA [29] enforces unique translatability from class proxies, encouraging same-subcategory samples to cluster in the embedding space. Despite their effectiveness, these methods mainly derive supervision from category labels and discriminative relations, which provides limited guidance for explicitly modeling the appearance distributions behind fine-grained categories. As a result, subtle visual patterns that are shared or comparable across categories may not be sufficiently captured. In contrast, GAPan learns category-specific appearance priors from feature distributions within a shared latent space. By leveraging these priors to generate appearance-aware anchors for supervision, GAPan encourages embeddings

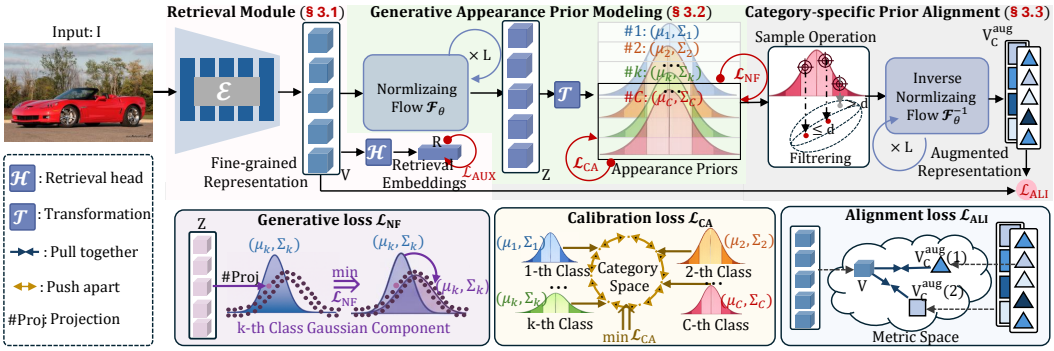

Figure 2: Detailed illustration of **Generative Appearance Prior alignment network**. See §3 for more details.

to capture fine-grained appearance cues while maintaining discrimination for retrieving both seen and unseen categories.

Normalizing Flow. Normalizing flows [8; 43] are a class of powerful generative models that transform complex data distributions into simple, tractable ones via invertible and differentiable mappings. They have attracted considerable attention due to their flexibility and precise likelihood estimation in high-dimensional settings. For instance, Glow [13] employs invertible 1×1 convolutions and exact log-likelihood optimization for high-quality image generation. Beyond generative modeling, normalizing flows also demonstrate strong performance in classification tasks. STG-NF [19] learns the data distribution and scores the samples according to their likelihood, while TWGC [22] highlights their robustness and interpretability compared to standard feed-forward models. Moreover, Flow-OOD [15] demonstrates their effectiveness in out-of-distribution detection. Unlike these works, GAPan leverages normalizing flows to model explicit category-specific appearance priors from instance features and further exploits the inverse flow to generate appearance-aware anchors for distribution-level supervision, complementing cross-category discrimination.

3 Method

The core of GAPan (Fig. 2) lies in shifting the supervisory paradigm from category prediction to appearance modeling. The framework integrates a retrieval backbone with two flow-driven modules:

- Forward Flow: Generative Appearance Prior Modeling maps instance features into a unified latent manifold, where its invertible property distills diverse observations into explicit category-specific Gaussian distributions while retaining all visual details.
- Reverse Flow: Category-specific Prior Alignment samples from high-density regions of these priors to generate appearance-aware anchors that provide distribution-level supervision, guiding retrieval embeddings toward underlying category-specific appearance characteristics.

3.1 Retrieval Module

The retrieval module is responsible for extracting visual representations and mapping them into a discriminative embedding space. Specifically, given an input image $\mathrm{I} \in \mathbb{R}^{3\times \mathrm{H}\times \mathrm{W}}$, the retrieval network $\mathcal{E}(\cdot)$ first extracts a high-dimensional feature embedding $\mathrm{V} = \mathcal{E}(\mathrm{I}) \in \mathbb{R}^{\mathrm{C}}$. To ensure a fair comparison with existing works [25; 26], we use a lightweight retrieval head $\mathcal{H}(\cdot)$ to project V into a compact C'-dimensional embedding space, yielding the final retrieval embedding $\mathrm{R} = \mathcal{H}(\mathrm{V}) \in \mathbb{R}^{\mathrm{C}'}$, where $\mathrm{C}' = 128$. Note that GAPan introduces no additional computational overhead at inference.

3.2 Generative Appearance Prior Modeling

To address the limited capacity of label-driven supervision to capture cross-category appearance comparability, we propose a generative appearance prior modeling module that learns category-specific priors in a shared latent space. This module utilizes *the forward transformation of normalizing flows* to map instance features into a unified latent manifold. Its invertibility ensures an information-

preserving transformation, distilling diverse observations into explicit category-specific Gaussian distributions without losing the subtle details typical of discriminative compression.

Feature Mapping via Normalizing Flows. To explicitly model appearance priors, we employ an invertible normalizing flow $\mathcal{F}_\theta(\cdot)$ to map instance features V into a latent space Z. This lossless transformation captures complex distributions while preserving fine-grained information. Specifically, we adopt a modified RealNVP architecture [3], where the transformation $\mathrm{Z} = \mathcal{F}_\theta(\mathrm{V})$ is achieved through L iterative coupling layers. Each layer partitions the input into $[\mathrm{Z}_1^{(\mathrm{l}-1)}, \mathrm{Z}_2^{(\mathrm{l}-1)}]$ and applies an affine transformation to one partition conditioned on the other:

$$\mathrm{Z}_1^{(\mathrm{l})} = \mathrm{Z}_1^{(\mathrm{l}-1)}, \quad \mathrm{Z}_2^{(\mathrm{l})} = \mathrm{Z}_2^{(\mathrm{l}-1)} \odot \exp\left(\mathrm{s}^{(\mathrm{l})}(\mathrm{Z}_1^{(\mathrm{l}-1)})\right) + \mathrm{t}^{(\mathrm{l})}(\mathrm{Z}_1^{(\mathrm{l}-1)}), \tag{1}$$

where $\mathrm{Z}^{(0)} = \mathrm{V}$ and $\mathrm{Z}^{(\mathrm{L})} = \mathrm{Z} = \mathcal{F}_\theta(\mathrm{V})$. Here, $\mathrm{s}^{(\mathrm{l})}(\cdot)$ and $\mathrm{t}^{(\mathrm{l})}(\cdot)$ denote the scale and translation networks in the l-th coupling layer, respectively, and both are implemented as MLPs. By stacking coupling layers with alternating partitions, the flow progressively maps complex feature distributions into a tractable latent space, where appearance priors can be effectively modeled.

Generative Probabilistic Modeling. Within the unified latent manifold established by the forward flow, we characterize category-specific appearance priors using class-conditional Gaussian distributions. Each category k is assigned a Gaussian distribution:

$$\mathrm{P}_\mathrm{Z}(\mathrm{Z} \mid \mathrm{k}) = \mathcal{N}(\mathrm{Z} \mid \mu_\mathrm{k}, \Sigma_\mathrm{k}), \tag{2}$$

where $\mu_k \in \mathbb{R}^C$ and $\Sigma_k \in \mathbb{R}^C$ denote the mean vector and the diagonal covariance matrix of the Gaussian distribution for the k-th category, respectively. This formulation represents the appearance prior of each category as a Gaussian distribution that captures rich intra-category variations.

Using the change-of-variables formula, we connect the latent appearance prior to the observed feature space. For an instance feature V with category y, its class-conditional density is given by

$$\mathrm{P}_\mathrm{V}(\mathrm{V} \mid \mathrm{k} = \mathrm{y}, \theta) = \mathcal{N}(\mathcal{F}_\theta(\mathrm{V}) \mid \mu_\mathrm{k}, \Sigma_\mathrm{k}) \cdot \left|\det \frac{\partial \mathcal{F}_\theta(\mathrm{V})}{\partial \mathrm{V}}\right|. \tag{3}$$

By coupling the flow transformation with the class-conditional Gaussian distributions, the model forms a unified probabilistic framework that organizes instance-level variations into category-specific appearance priors for fine-grained representation learning.

Normalizing Flow Optimization. To improve the accuracy of density modeling and discriminative ability of the learned appearance priors, a dual-objective function is employed. First, the generative loss $\mathcal{L}_\mathrm{NF}$ is formulated as the negative log-likelihood (NLL) under Eq. (3):

$$\mathcal{L}_\mathrm{NF} = -\frac{1}{|\mathcal{B}|} \sum_{(\mathrm{V_i}, \mathrm{y_i}) \in \mathcal{B}} \log \mathrm{P}_\mathrm{V}(\mathrm{V_i} \mid \mathrm{k} = \mathrm{y_i}, \theta), \tag{4}$$

where $\mathcal{B}$ and $|\mathcal{B}|$ denote a mini-batch and its corresponding size, respectively. Minimizing the NLL jointly optimizes the flow parameters θ and Gaussian parameters $\{\mu_\mathrm{k}, \Sigma_\mathrm{k}\}_{\mathrm{k}=1}^{\mathrm{K}}$, encouraging each instance feature to lie in a high-density region of its class-conditional distribution, where K denotes the number of seen categories.

To preserve discriminative ability, we introduce a posterior calibration constraint $\mathcal{L}_\mathrm{CA}$ that encourages category distributions to remain separable. The posterior probability of an instance belonging to class k is computed by normalizing its likelihood across all class-conditional distributions:

$$\mathrm{P}(\mathrm{k} \mid \mathrm{V}) = \frac{\mathcal{N}(\mathcal{F}_\theta(\mathrm{V}) \mid \mu_\mathrm{k}, \Sigma_\mathrm{k})}{\sum_{\mathrm{j}=1}^{\mathrm{K}} \mathcal{N}(\mathcal{F}_\theta(\mathrm{V}) \mid \mu_\mathrm{j}, \Sigma_\mathrm{j})}. \tag{5}$$

$\mathcal{L}_\mathrm{CA}$ minimizes the cross-entropy between this posterior and the corresponding ground-truth label $\mathrm{y_i}$:

$$\mathcal{L}_\mathrm{CA} = -\frac{1}{|\mathcal{B}|} \sum_{(\mathrm{V_i}, \mathrm{y_i}) \in \mathcal{B}} \log \mathrm{P}(\mathrm{k} = \mathrm{y_i} \mid \mathrm{V_i}). \tag{6}$$

While the NLL increases the likelihood under the target category distribution, the posterior calibration loss normalizes the likelihood across categories and explicitly promotes inter-category separation. Together, these objectives couple the flow transformation with the class-conditional Gaussian distributions, producing appearance priors that faithfully characterize category-specific visual patterns and remain discriminative for fine-grained retrieval.

3.3 Category-specific Prior Alignment

To directly transfer appearance priors into the retrieval embedding space, we introduce a category-specific prior alignment module that exploits *the inverse property of normalizing flows*, sampling from high-density regions of the learned Gaussian priors to generate appearance-aware anchors in the original feature space. These anchors serve as distribution-level supervisory signals, guiding the retrieval model to align instance embeddings with the underlying appearance characteristics of their corresponding categories.

For each class k, this module samples $\mathrm{N_S}$ latent vectors $\mathrm{S_{k,i}} \in \mathbb{R}^{\mathrm{C}}$ from the corresponding Gaussian component $\mathcal{N}(\mu_{\mathrm{k}}, \Sigma_{\mathrm{k}})$ via rejection sampling. These latent vectors are then transformed back into the feature space via the inverse flow $\mathcal{F}_\theta^{-1}(\cdot)$, producing synthesized feature anchors $\mathrm{V}_{\mathrm{k,i}}^{\mathrm{aug}} \in \mathbb{R}^{\mathrm{C}}$:

$$\mathrm{V}_{\mathrm{k,i}}^{\mathrm{aug}} = \mathcal{F}_\theta^{-1}(\mathrm{S_{k,i}}), \quad \mathrm{S_{k,i}} \sim \mathcal{N}(\mu_{\mathrm{k}}, \Sigma_{\mathrm{k}}), \quad \text{s.t.} \quad \sqrt{(\mathrm{S_{k,i}} - \mu_{\mathrm{k}})^\top \Sigma_{\mathrm{k}}^{-1} (\mathrm{S_{k,i}} - \mu_{\mathrm{k}})} \leq \mathrm{d}. \tag{7}$$

where $\mathrm{d} > 0$ is a threshold controlling the sampling region within the Gaussian distribution. The threshold d governs the trade-off between sampling reliability and diversity: a smaller d restricts sampling to high-probability regions, yielding typical feature anchors that capture reliable category-specific appearance cues, whereas a larger d explores a wider region to provide more diverse anchors.

For the m-th training instance with label $\mathrm{y_m}$, generated features from the same category, *i.e.*, $\{\mathrm{V}_{\mathrm{y_m,i}}^{\mathrm{aug}}\}_{\mathrm{i}=1}^{\mathrm{N_S}}$, are treated as positive anchors, while those from other categories serve as negatives. These generated features are projected into the retrieval space by the same head $\mathcal{H}(\cdot)$ used for real instance features, so that each retrieval embedding $\mathrm{R_m}$ can be aligned with its category-matched synthesized anchors via the prior-guided alignment loss:

$$\mathcal{L}_{\mathrm{ALI}} = -\frac{1}{|\mathcal{B}| \times \mathrm{N_S}} \sum_{\mathrm{m}=1}^{|\mathcal{B}|} \sum_{\mathrm{i}=1}^{\mathrm{N_S}} \log \frac{\exp\Big(\mathrm{sim}(\mathrm{R_m}, \mathcal{H}(\mathrm{V}_{\mathrm{y_m,i}}^{\mathrm{aug}}))/\tau\Big)}{\sum_{\mathrm{c}=1}^{\mathrm{K}} \sum_{\mathrm{j}=1}^{\mathrm{N_S}} \exp\Big(\mathrm{sim}(\mathrm{R_m}, \mathcal{H}(\mathrm{V}_{\mathrm{c,j}}^{\mathrm{aug}}))/\tau\Big)}, \tag{8}$$

where $\mathrm{sim}(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature parameter. This loss injects learned appearance priors into the retrieval embedding space, encouraging representations to be organized around appearance cues rather than category semantics. Consequently, the alignment objective promotes capturing transferable appearance cues, which improves generalization to unseen categories.

3.4 Overall Training Objective

To ensure stable convergence and generative-discriminative alignment, GAPan adopts a progressive two-stage training protocol separated by a threshold epoch T.

Auxiliary Contrastive Learning. During the initial stage, the retrieval model is preconditioned using an auxiliary contrastive loss $\mathcal{L}_{\mathrm{AUX}}$ to establish a robust feature manifold. For a mini-batch $\mathcal{B}$ containing N categories with two instances per category, the batch size is $|\mathcal{B}| = 2\mathrm{N}$. For each instance i, let $p(i)$ denote the index of the other instance of the same category in $\mathcal{B}$ (with $p(i) \neq i$). The loss is defined as:

$$\mathcal{L}_{\mathrm{AUX}} = -\frac{1}{|\mathcal{B}|} \sum_{\mathrm{i}=1}^{|\mathcal{B}|} \log \frac{\exp\big(\mathrm{sim}(\mathrm{R_i}, \mathrm{R_{p(i)}})/\tau\big)}{\sum_{\mathrm{m}=1,\mathrm{m}\neq\mathrm{i}}^{|\mathcal{B}|} \exp(\mathrm{sim}(\mathrm{R_i}, \mathrm{R_m})/\tau)}, \tag{9}$$

where τ is the same temperature as in Eq. (8).

Progressive Training Schedule. The optimization process is partitioned into a warmup phase and a joint optimization phase based on the current epoch t:

- **Phase I: Warmup** ($\mathrm{t} \leq \mathrm{T}$)**:** During the warmup phase, only the retrieval model is optimized with the auxiliary contrastive loss, providing stable feature representations for subsequent flow-based prior modeling. The total objective is $\mathcal{L} = \mathcal{L}_{\mathrm{AUX}}$.
- **Phase II: Joint Optimization** ($\mathrm{t} > \mathrm{T}$)**:** After epoch T, the normalizing flow and the retrieval model are jointly optimized by combining contrastive supervision, probabilistic prior modeling, calibration, and prior-based alignment:

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{AUX}} + \alpha\mathcal{L}_{\mathrm{NF}} + \beta\mathcal{L}_{\mathrm{CA}} + \gamma\mathcal{L}_{\mathrm{ALI}}, \tag{10}$$

where α, β, and γ are hyperparameters that balance the contributions of the corresponding loss terms.

4 Experiments

4.1 Experimental Setup

Datasets. CUB-200-2011 [2] consists of 200 bird species. We use the first 100 categories (5,864 images) for training and the remaining 100 categories (5,924 images) for testing. Stanford Cars [17] includes 196 car models. Similarly, we use the first 98 categories (8,054 images) for training and the remaining 98 categories (8,131 images) for testing. Stanford Online Products (SOP) [30] is divided into 11,318 categories (59,551 images) for training and 11,316 categories (60,502 images) for testing. *This split ensures **no category overlap** between training and testing sets, where all testing categories are strictly unseen during training to evaluate cross-category generalization.*

Implementation Details. Our retrieval model is implemented on top of ViT-B/16 [4] pre-trained on ImageNet-21K [27]. Input images are resized to 256×256 and randomly cropped to 224×224 during training. We adopt stochastic gradient descent with an initial learning rate of 1×10^{-5}, momentum of 0.9, and weight decay of 1×10^{-4}, using a batch size of 270 on two NVIDIA A6000 GPUs. The learning rate follows an exponential decay schedule with a decay factor of 0.9 every five epochs over a total of 50 training epochs. The scale and translation networks of normalizing flow are multi-layer perceptrons with a hidden dimension of 512. τ is set to 0.09 in Eqs. (8) and (9).

Evaluation protocol. We evaluate retrieval performance using Recall@K with cosine distance, following the standard protocol in prior work [31]. Specifically, for each query image, the model retrieves the top-K most similar images. A score of 1 is assigned if at least one positive image appears within the top-K results, and 0 otherwise. The final Recall@K is obtained by averaging these scores across all queries in the test set.

4.2 Ablation Experiments

We evaluate the contribution of each component in GAPan on the CUB-200-2011 dataset. As shown in Tab. 1, $\mathcal{L}_{\mathrm{AUX}}$ provides the basic retrieval supervision and establishes the embedding space for appearance prior modeling. Adding $\mathcal{L}_{\mathrm{NF}}$ improves Recall@1 from 82.6% to 84.7%, showing that coupling the flow transformation with class-conditional Gaussian distributions helps organize instance-level variations into category-specific appearance priors. With $\mathcal{L}_{\mathrm{CA}}$, the performance further increases to 85.3%, indicating that posterior calibration preserves discriminative ability by encouraging better separation among category-specific Gaussian distributions. The largest gain comes from $\mathcal{L}_{\mathrm{ALI}}$, which improves Recall@1 by 3.8%. This loss samples feature anchors from high-density regions of the learned category distributions and uses them as additional positive anchors in a contrastive objective. Instead of directly forcing each instance to match a single prototype, $\mathcal{L}_{\mathrm{ALI}}$ aligns instance embeddings with category-specific appearance priors while maintaining separation from other categories. This appearance-prior-guided supervision reduces reliance on sparse category labels and leads to stronger generalization for fine-grained retrieval.

Table 1: Comparison on CUB-200-2011 dataset using different combinations of constraints.

$\mathcal{L}_{\mathrm{AUX}}$	$\mathcal{L}_{\mathrm{NF}}$	$\mathcal{L}_{\mathrm{CA}}$	$\mathcal{L}_{\mathrm{ALI}}$	Recall@1
✓	✗	✗	✗	82.6%
✓	✓	✗	✗	84.7%
✓	✓	✓	✗	85.3%
✓	✓	✓	✓	89.1%

4.3 Comparison with the State-of-the-Art Methods

Fine-grained Image Retrieval. As shown in Tab. 2, GAPan achieves state-of-the-art performance on two widely-used fine-grained retrieval benchmarks, CUB-200-2011 and Stanford Cars 196. It obtains Recall@1 scores of 89.1% and 92.6%, outperforming the previous best method, LaFG [42], by absolute margins of 1.9% and 1.1%, respectively. Existing competitive methods, such as DDML [25] and SEE [18], mainly rely on category-level supervision to learn discriminative embeddings, which may limit their ability to capture visual regularities that transfer to unseen categories. In contrast, GAPan distills diverse instance observations into explicit appearance priors, enabling the model to learn visual patterns beyond category labels. Its improvement over LaFG, which incorporates external VLM knowledge, further demonstrates the effectiveness of appearance-prior-guided alignment learned directly from instance distributions.

Generalization on Coarse-grained Retrieval. To further evaluate the generalization capacity of GAPan, we conduct experiments on the large-scale Stanford Online Products (SOP) dataset. Even in

Table 2: Compared with competitive methods on CUB-200-2011, Stanford Cars 196 and Stanford Online Products datasets. "Arch" denotes the backbone architecture. "R50" and "ViT" denote the ResNet-50 [7] and Vision Transformer [4] backbones, respectively.

Method	Arch	CUB-200-2011				Stanford Cars 196				Stanford Online Products			
		1	2	4	8	1	2	4	8	1	10	100	1000
$\text{CBML}_{\text{TPAMI23}}$ [10]	R50	69.9	80.4	87.2	92.5	88.1	92.6	95.4	97.4	79.9	91.5	96.5	98.9
$\text{NIR}_{\text{CVPR22}}$ [29]	R50	70.5	80.6	-	-	89.1	93.4	-	-	80.4	91.4	-	-
$\text{HSE}_{\text{ICCV23}}$ [49]	R50	70.6	80.1	87.1	-	89.6	93.8	96.0	-	80.0	91.4	96.3	-
$\text{IDML}_{\text{TPAMI24}}$ [34]	R50	70.7	80.2	-	-	90.6	94.5	-	-	81.5	-	-	-
HIST $_{\text{CVPR22}}$ [20]	R50	71.4	81.1	88.1	-	89.6	93.9	96.4	-	81.4	92.0	96.7	-
$\text{PNCA++}_{\text{ECCV20}}$ [32]	R50	72.2	82.0	89.2	93.5	90.1	94.5	97.0	98.4	81.4	92.4	96.9	99.0
$\text{DIML}_{\text{TPAMI24}}$ [50]	ViT	76.7	-	-	-	80.7	-	-	-	79.5	-	-	-
$\text{DFML-PA}_{\text{CVPR23}}$ [33]	ViT	79.1	86.8	-	-	89.5	93.9	-	-	84.2	93.8	-	-
DVA $_{\text{AAAI26}}$ [9]	ViT	84.9	90.6	94.5	96.7	90.7	94.8	97.1	98.4	-	-	-	-
$\text{DPHM}_{\text{PR25}}$ [47]	ViT	85.5	91.3	94.6	96.6	84.1	90.5	94.5	97.1	84.8	94.5	98.1	99.4
$\text{HypViT}_{\text{CVPR22}}$ [5]	ViT	85.6	91.4	94.8	96.7	89.2	94.1	96.7	98.1	85.9	94.9	98.1	99.5
$\text{HIER}_{\text{CVPR23}}$ [11]	ViT	85.7	91.3	94.4	-	88.3	93.2	96.1	-	86.1	95.0	98.0	-
$\text{SEE}_{\text{IJCAI25}}$ [18]	ViT	85.8	91.4	94.6	-	88.8	93.8	96.4	-	86.3	95.0	98.2	-
$\text{DDML}_{\text{AAAI25}}$ [25]	ViT	86.0	91.7	95.2	96.8	89.5	94.2	96.8	98.2	86.1	95.1	98.2	99.5
$\text{VPTSP-G}_{\text{ICLR24}}$ [26]	ViT	86.6	91.7	94.8	-	87.7	93.3	96.1	-	84.4	93.6	97.3	-
$\text{LaFG}_{\text{ARXIV25}}$ [42]	ViT	87.2	92.4	95.2	97.0	91.5	94.6	96.6	98.5	87.1	95.8	98.5	99.5
Our GAPan	ViT	**89.1**	**93.5**	**95.3**	**97.1**	**92.6**	**95.7**	**97.6**	**98.7**	**88.4**	**96.6**	**98.9**	**99.6**

this coarse-grained setting, GAPan achieves a Recall@1 of 88.4%, surpassing recent competitive methods such as DPHM [47] (+3.6%) and LaFG [42] (+1.3%). SOP poses a different challenge from fine-grained benchmarks, as its coarse labels contain larger intra-class variation and provide weaker fine-grained semantic cues. By using category labels to organize instance-level distributions, GAPan distills reliable appearance priors that remain effective across different levels of category granularity. The consistent improvements across both fine-grained and coarse-grained datasets show that appearance-prior-guided supervision, rather than relying only on discrete category labels, improves cross-category comparability and yields more discriminative and generalizable embeddings.

4.4 Further Analysis

Effect on the depth of normalizing flow. We further investigate the sensitivity of GAPan to the depth of the normalizing flow by varying the number of coupling layers L from 1 to 16. As summarized in Tab. 3, Recall@1 on CUB-200-2011 first increases with the flow depth and reaches the best performance of 89.1% at $\mathrm{L} = 8$, then drops to 87.8% when $\mathrm{L} = 16$. When the flow is too shallow, its limited transformation capacity makes it difficult to map complex instance features into a structured latent space for appearance prior modeling. In contrast, an overly deep flow introduces more learnable parameters and may overfit the relatively limited fine-grained training data. These results suggest that an appropriate flow depth allows the model to capture complex feature distributions without introducing unnecessary optimization difficulty.

Table 3: Retrieval accuracy on CUB-200-2011 of model trained with different number of flow layers L.

Number L	1	2	4	8	16
Recall@1	84.7%	86.3%	87.9%	89.1%	87.8%

Sensitivity on the number of synthesized anchors. As shown in Tab. 4, we evaluate the effect of the number of generated anchors $\mathrm{N_S}$ in GAPan. Recall@1 increases from 87.6% to 89.1% as $\mathrm{N_S}$ grows from 1 to 6, showing that more anchors provide richer supervision from the learned category-specific appearance priors. However, further increasing $\mathrm{N_S}$ to 8 reduces performance to 87.8%. This suggests that excessive generated anchors may introduce redundant constraints and weaken the preservation of instance-level variations. These results suggest that an appropriate number of generated anchors strengthens prior-guided supervision without suppressing instance-level variation.

Table 4: Accuracy on CUB-200-2011 for models trained with different numbers of sampling anchors.

Number $\mathrm{N_S}$	1	2	4	6	8
Recall@1	87.6%	88.0%	88.4%	89.1%	87.8%

Impact of sampling threshold in Eq. (7)**.** As shown in Tab. 5, we evaluate the effect of the sampling threshold d in Eq. (7). Recall@1 reaches the best performance of 89.1% at $\mathrm{d} = 1.0$. When d is too small, such as $\mathrm{d} = 0.2$, sampling is restricted to a narrow high-density region around the center of the

Gaussian distribution, producing reliable but less diverse anchors and thus limiting the supervision provided by the learned appearance priors. When d is too large, such as $\mathrm{d} = 2.0$, sampling covers a broader region of the Gaussian distribution and may include less representative anchors, which can introduce noisy supervision. The optimal threshold $\mathrm{d} = 1.0$ provides a better balance between anchor reliability and diversity, leading to stronger fine-grained retrieval performance.

Table 5: Accuracy on CUB-200-2011 for models trained with diverse threshold d for generated anchors.

Threshold d	0.2	0.5	1	1.5	2.0
Recall@1	86.1%	87.9%	89.1%	88.3%	87.4%

Effect of warm-up epochs. We examine the impact of the warm-up strategy by varying the number of initial warm-up epochs T. As shown in Tab. 6, Recall@1 reaches the best performance of 89.1% at $\mathrm{T} = 5$. Without warm-up, the normalizing flow is optimized on unstable early-stage features, which weakens appearance prior modeling. When T is too small, the embedding space is not sufficiently stable for the flow to estimate reliable category distributions. In contrast, an overly long warm-up period leaves fewer epochs for optimizing the flow and the appearance-prior-guided supervision, leading to slightly lower performance. These results suggest that a moderate warm-up period provides sufficiently stable features for appearance prior modeling while leaving enough training time for global optimization.

Table 6: Retrieval accuracy on CUB-200-2011 with different warm-up epochs T.

Epoch T	0	2	5	8	10
Recall@1	88.2%	88.8%	89.1%	89.0%	88.7%

Hyperparameter analysis. We conduct sensitivity analyses of the hyperparameters in Eq. (10), with the results shown in Fig. 3. The performance of GAPan varies with the weights α, β, and γ, showing that these objectives need to be properly balanced. When these weights are too small, the corresponding objectives, including distribution modeling, posterior calibration, and feature sampling contrastive supervision, contribute insufficiently to training. In contrast, overly large weights may overemphasize the auxiliary objectives and weaken retrieval discrimination. Based on these observations, we therefore set $\alpha = 0.5$, $\beta = 0.3$, and $\gamma = 0.4$ as the default values.

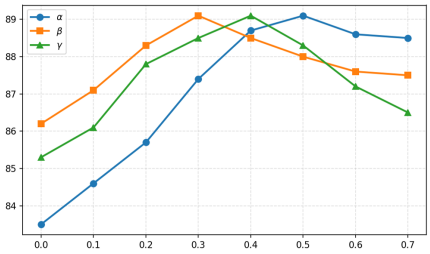

Figure 3: Hyperparameter analysis of α, β and γ in Eq. (10).

Effectiveness of generative appearance priors. As shown in Tab. 7, we compare GAPan with point-based representation strategies, including K-means clustering and prototypical learning, to evaluate the effectiveness of generative appearance priors. K-means represents feature structures with discrete centroids, which may smooth out fine-grained variations within each category. Prototypical learning uses a fixed prototype for each category, making it difficult to describe diverse intra-category appearances. In contrast, GAPan models each category as a probabilistic distribution in a flow-based latent space and generates multiple appearance anchors from high-density regions of the learned distributions. This distribution-based formulation provides richer supervision than point-based representations, leading to better cross-category comparability and stronger generalization.

Table 7: Performance comparison between GAPan and point-based representation strategies, including K-means clustering and prototypical learning, on CUB-200-2011.

Method	K-means	Proto	GAPan
Recall@1	83.9%	85.4%	89.1%

Importance of normalizing flows. We compare GAPan with an encoder-decoder (ED) baseline implemented as a multi-layer MLP, using a comparable parameter budget, to evaluate the contribution of the flow-based architecture to appearance prior modeling. As shown in Tab. 8, replacing the normalizing flow with ED reduces Recall@1 from 89.1% to 84.9%. This indicates that a standard non-invertible mapping is less effective in connecting the observed feature space with the latent category distributions. In contrast, the normalizing flow provides an invertible transformation and enables exact likelihood estimation through the change-of-variables formula, allowing instance features and class-conditional Gaussian distributions to be coupled in a unified probabilistic framework. This coupling keeps the sampled anchors consistent with the learned category distributions, providing more effective appearance-prior-guided supervision for FGIR.

Table 8: Retrieval accuracy on CUB-200-2011 for models trained with either a normalizing flow (NF) or an encoder–decoder (ED) architecture implemented as a multi-layer MLP.

Method	ED	NF ($\mathcal{F}_\theta$)
Recall@1	84.9%	89.1%

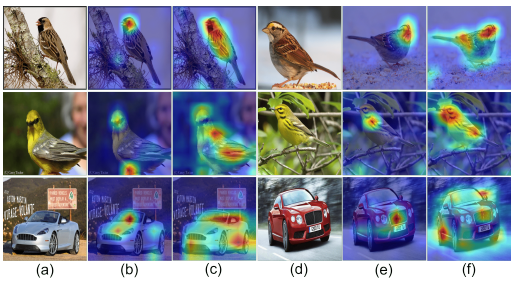

Figure 4: Illustration of class activation maps. (a) and (d) denote the input images, (b) and (e) show the activation maps produced by the baseline, and (c) and (f) present those generated by GAPan.

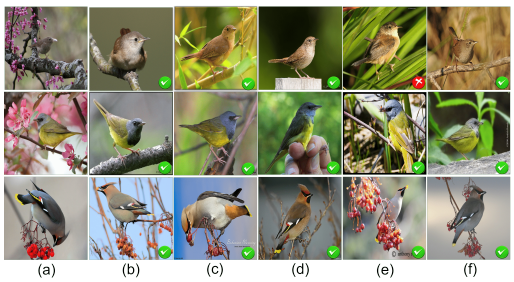

Figure 5: Examples of top-5 retrieval results on CUB-200-2011 by GAPan. (a) denotes the query image, and (b)–(f) show the retrieved images in descending order of similarity.

Feature Representation Analysis. Since the learned appearance priors are optimized in a latent space and cannot be directly visualized, we interpret their effect indirectly using class activation maps and retrieval examples, which respectively show the attended visual regions and the resulting cross-instance retrieval behavior. As shown in Fig. 4, the baseline mainly activates the most discriminative local regions, which is consistent with its reliance on category-level supervision. In contrast, GAPan produces broader and more complete activation responses over object regions, covering additional visual cues such as body parts, texture, and contour. This indicates that the proposed appearance priors encourage the model to capture category-specific appearance patterns beyond isolated discriminative regions. Consistent with this broader visual perception, the retrieval examples in Fig. 5 show that GAPan retrieves images with similar fine-grained appearance despite variations in pose and background. Given a query image, GAPan retrieves visually similar birds with consistent fine-grained characteristics, even when pose, scale, or background varies. This suggests that the learned embeddings are guided by transferable appearance cues rather than only by category-specific decision boundaries. Together, the activation maps and retrieval results show that GAPan improves representation learning by aligning instance embeddings with learned appearance priors, leading to more complete object perception and stronger generalization to unseen fine-grained categories.

5 Conclusion and Discussion

In this paper, we introduced GAPan, a framework that leverages the invertible nature of normalizing flows to establish a shared latent density space, shifting the learning paradigm from category prediction toward appearance modeling. Through the forward flow, GAPan models diverse instance observations with explicit category-specific Gaussian priors, while the reverse flow generates appearance-aware anchors to provide distribution-level supervision. This bidirectional generative framework effectively aligns retrieval embeddings with transferable appearance characteristics across categories. Extensive evaluations demonstrate that GAPan consistently outperforms existing methods and exhibits superior generalization to novel subcategories across both fine- and coarse-grained benchmarks.

Broader Impact. Fine-grained retrieval is pivotal for specialized domains such as biodiversity monitoring, precision agriculture, and e-commerce. GAPan advances FGIR by shifting from sparse label-based supervision to distribution-aware appearance modeling. This paradigm is particularly valuable for low-resource or open-set scenarios where new or rare categories frequently emerge; by capturing underlying visual priors, our model reduces the dependency on exhaustive manual annotations and improves generalization to unseen classes. This capability supports more robust and scalable visual analysis in real-world applications where data labeling is often a bottleneck.

Limitations. The deployment of FGIR systems requires careful validation of data integrity. When the training distribution is long-tailed or lacks sufficient visual diversity, the learned appearance priors tend to inherit these biases, leading to degraded and inconsistent performance across domains. In real-world applications, such as endangered species monitoring or commercial recommendation, we advocate using GAPan as a decision-support tool rather than an autonomous system, with human-in-the-loop verification to mitigate risks arising from distribution shifts and domain gaps.

Acknowledgements. This work was partially supported by ARC DE240100105, DP230101196, IE250100108, UOQ2301-010OPX and ARC Industrial Transformation Research Hubs IH230100013.

References

[1] Kenan E. Ak, Ashraf A. Kassim, Joo-Hwee Lim, and Jo Yew Tham. Learning attribute representations with localization for flexible fashion search. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7708–7717. Computer Vision Foundation / IEEE Computer Society, 2018.

[2] Steve Branson, Grant Van Horn, Serge J. Belongie, and Pietro Perona. Bird species categorization using pose normalized deep convolutional nets. *CoRR*, abs/1406.2952, 2014.

[3] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[5] Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan V. Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 7399–7409. IEEE, 2022.

[6] Takuya Furusawa. Mean field theory in deep metric learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778, 2016.

[8] Or Hirschorn and Shai Avidan. Normalizing flows for human pose anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13545–13554, 2023.

[9] Xin Jiang, Meiqi Cao, Hao Tang, Fei Shen, and Zechao Li. Fine-grained image retrieval via dual-vision adaptation. *arXiv preprint arXiv:2506.16273*, 2025.

[10] Shichao Kan, Zhiquan He, Yigang Cen, Yang Li, Vladimir Mladenovic, and Zhihai He. Contrastive bayesian analysis for deep metric learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(6):7220–7238, 2023.

[11] Sungyeon Kim, Boseung Jeong, and Suha Kwak. HIER: metric learning beyond class labels via hierarchical regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 19903–19912. IEEE, 2023.

[12] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Embedding transfer with label relaxation for improved metric learning. In *CVPR*, pages 3967–3976. Computer Vision Foundation / IEEE, 2021.

[13] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[14] Michael Kirchhof, Karsten Roth, Zeynep Akata, and Enkelejda Kasneci. A non-isotropic probabilistic take on proxy-based deep metric learning. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVI*, Lecture Notes in Computer Science, pages 435–454. Springer, 2022.

[15] Polina Kirichenko, Pavel Izmailov, and Andrew G Wilson. Why normalizing flows fail to detect out-of-distribution data. *Advances in neural information processing systems*, 33:20578–20589, 2020.

[16] ByungSoo Ko, Geonmo Gu, Han-Gyu Kim, and ByungSoo Ko. Learning with memory-based virtual classes for deep metric learning. In *ICCV*, pages 11772–11781. IEEE, 2021.

[17] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV Workshops 2013, Sydney, Australia, December 1-8, 2013*, pages 554–561, 2013.

[18] Binh Minh Le and Simon S. Woo. SEE: spherical embedding expansion for improving deep metric learning (extended abstract). In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*, pages 10906–10911. ijcai.org, 2025.

[19] Jiarui Lei, Xiaobo Hu, Yue Wang, and Dong Liu. Pyramidflow: High-resolution defect contrastive localization using pyramid normalizing flow. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14143–14152, 2023.

[20] Jongin Lim, Sangdoo Yun, Seulki Park, and Jin Young Choi. Hypergraph-induced semantic tuplet loss for deep metric learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 212–222. IEEE, 2022.

[21] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 1096–1104. IEEE Computer Society, 2016.

[22] Radek Mackowiak, Lynton Ardizzone, Ullrich Kothe, and Carsten Rother. Generative classifiers as a basis for trustworthy image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2971–2981, 2021.

[23] Bálint Máté, Samuel Klein, Tobias Golling, and François Fleuret. Flowification: Everything is a normalizing flow. *Advances in Neural Information Processing Systems*, 35:35478–35489, 2022.

[24] Olga Moskvyak, Frédéric Maire, Feras Dayoub, and Mahsa Baktashmotlagh. Keypoint-aligned embeddings for image retrieval and re-identification. In *Winter Conference on Applications of Computer Vision*, pages 676–685. IEEE, 2021.

[25] Jinhee Park, Jisoo Park, Dagyeong Na, and Junseok Kwon. Deep disentangled metric learning. In Toby Walsh, Julie Shah, and Zico Kolter, editors, *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 19830–19838. AAAI Press, 2025.

[26] Li Ren, Chen Chen, Liqiang Wang, and Kien A. Hua. Learning semantic proxies from visual prompts for parameter-efficient fine-tuning in deep metric learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[27] Tal Ridnik, Emanuel Ben Baruch, Asaf Noy, and Lihi Zelnik. Imagenet-21k pretraining for the masses. In Joaquin Vanschoren and Sai-Kit Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021.

[28] Karsten Roth, Timo Milbich, Björn Ommer, Joseph Paul Cohen, and Marzyeh Ghassemi. Simultaneous similarity-based self-distillation for deep metric learning. In Marina Meila and Tong Zhang, editors, *Proceedings of Machine Learning Research*, volume 139 of *Proceedings of Machine Learning Research*, pages 9095–9106. PMLR, 2021.

[29] Karsten Roth, Oriol Vinyals, and Zeynep Akata. Non-isotropy regularization for proxy-based deep metric learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 7410–7420. IEEE, 2022.

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823. IEEE Computer Society, 2015.

[31] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012. IEEE Computer Society, 2016.

[32] Eu Wern Teh, Terrance DeVries, Graham W. Taylor, and Graham. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *ECCV*, volume 12369 of *Lecture Notes in Computer Science*, pages 448–464. Springer, 2020.

[33] Chengkun Wang, Wenzhao Zheng, Junlong Li, Jie Zhou, and Jiwen Lu. Deep factorized metric learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 7672–7682. IEEE, 2023.

[34] Chengkun Wang, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Introspective deep metric learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(4):1964–1980, 2024.

[35] Shijie Wang, Jianlong Chang, Haojie Li, Zhihui Wang, Wanli Ouyang, and Qi Tian. Learning to parameterize visual attributes for open-set fine-grained retrieval. In *Annual Conference on Neural Information Processing Systems, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[36] Shijie Wang, Jianlong Chang, Haojie Li, Zhihui Wang, Wanli Ouyang, and Qi Tian. Open-set fine-grained retrieval via prompting vision-language evaluator. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 19381–19391. IEEE, 2023.

[37] Shijie Wang, Jianlong Chang, Zhihui Wang, Haojie Li, Wanli Ouyang, and Qi Tian. Fine-grained retrieval prompt tuning. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 2644–2652. AAAI Press, 2023.

[38] Shijie Wang, Jianlong Chang, Zhihui Wang, Haojie Li, Wanli Ouyang, and Qi Tian. Content-aware rectified activation for zero-shot fine-grained image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(6):4366–4380, 2024.

[39] Shijie Wang, Jian Shi, and Haojie Li. Adversarial reconstruction feedback for robust fine-grained generalization. In *IEEE/CVF International Conference on Computer Vision, ICCV 2025, Honolulu, HI, USA, October 19-25, 2025*, pages 3080–3090. IEEE, 2025.

[40] Shijie Wang, Zhihui Wang, Haojie Li, Jianlong Chang, Wanli Ouyang, and Qi Tian. Accurate fine-grained object recognition with structure-driven relation graph networks. *Int. J. Comput. Vis.*, 132(1):137–160, 2024.

[41] Shijie Wang, Zhihui Wang, Haojie Li, and Wanli Ouyang. Category-specific nuance exploration network for fine-grained object retrieval. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 2513–2521. AAAI Press, 2022.

[42] Shijie Wang, Xin Yu, Yadan Luo, Zijian Wang, Pengfei Zhang, and Zi Huang. Language-driven fine-grained retrieval. *CoRR*, abs/2512.06255, 2025.

[43] Yufei Wang, Renjie Wan, Wenhan Yang, Haoliang Li, Lap-Pui Chau, and Alex Kot. Low-light image enhancement with normalizing flow. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 2604–2612, 2022.

[44] Zhihui Wang, Shijie Wang, Shuhui Yang, Haojie Li, Jianjun Li, and Zezhou Li. Weakly supervised fine-grained image classification via guassian mixture model oriented discriminative learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9746–9755. Computer Vision Foundation / IEEE, 2020.

[45] Zhihui Wang, Shijie Wang, Pengbo Zhang, Haojie Li, Wei Zhong, and Jianjun Li. Weakly supervised fine-grained image classification via correlation-guided discriminative learning. In *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*, pages 1851–1860. ACM, 2019.

[46] Xiu-Shen Wei, Yang Shen, Xuhao Sun, Han-Jia Ye, and Jian Yang. A^2-net: Learning attribute-aware hash codes for large-scale fine-grained image retrieval. *Advances in Neural Information Processing Systems*, 34:5720–5730, 2021.

[47] Yunhao Xu, Zhentao Chen, and Junlin Hu. Deep metric learning in projected-hypersphere space. *Pattern Recognit.*, 161:111245, 2025.

[48] Shiyang Yan, Lin Xu, Xinyao Shu, Zhenyu Lu, and Jialie Shen. Lm-metric: Learned pair weighting and contextual memory for deep metric learning. *Pattern Recognit.*, 155:110722, 2024.

[49] Bailin Yang, Haoqiang Sun, Frederick W. B. Li, Zheng Chen, Jianlu Cai, and Chao Song. HSE: hybrid species embedding for deep metric learning. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 11013–11023. IEEE, 2023.

[50] Wenliang Zhao, Yongming Rao, Jie Zhou, and Jiwen Lu. DIML: deep interpretable metric learning via structural matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(4):2518–2532, 2024.

[51] Wenzhao Zheng, Chengkun Wang, Jiwen Lu, and Jie Zhou. Deep compositional metric learning. In *CVPR*, pages 9320–9329. Computer Vision Foundation / IEEE, 2021.

[52] Wenzhao Zheng, Borui Zhang, Jiwen Lu, and Jie Zhou. Deep relational metric learning. In *ICCV*, pages 12045–12054. IEEE, 2021.

[53] Xiawu Zheng, Rongrong Ji, Xiaoshuai Sun, Yongjian Wu, Feiyue Huang, and Yanhua Yang. Centralized ranking loss with weakly supervised localization for fine-grained object retrieval. In Jérôme Lang, editor, *IJCAI*, pages 1226–1233. ijcai.org, 2018.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We make the main claims clearly in the abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation of this paper in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer:[Yes]

   Justification: We discuss the experimental setup in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code and benchmark would be released after submission and review.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Details of training and testing are specified in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Experiments in this paper are conducted in multiple iterations, yielding stable results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computational cost and resources are specified in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conducted in this paper conforms in every respect with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The Broader Impacts are discussed in Section 5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: Our work does not pose risks requiring such safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All models and datasets used in this paper have been properly credited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLM is only used for writing.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.